Configuring Spanning Tree

This chapter describes how to configure Spanning-Tree Protocol, and how to use two Spanning-Tree features: Spanning-Tree Uplink Fast Switchover, and Spanning-Tree Protocol Backbone Fast Convergence.

Configuring Spanning-Tree Protocol

The Catalyst 5000 series switch uses Spanning-Tree Protocol (IEEE 802.1D bridge protocol) on all Ethernet, Fast Ethernet, Gigabit Ethernet, and Token Ring port-based VLANs. When you create fault-tolerant internetworks, you must have a loop-free path between all nodes in a network. In Spanning-Tree protocol, an algorithm calculates the best loop-free path throughout a Catalyst 5000 series switched network. The Catalyst 5000 series switches send and receive spanning-tree packets at regular intervals. The switches do not forward the packets, but use the packets to identify a loop-free path. The default configuration has all Spanning-Tree Protocols enabled.

For more information on Spanning-Tree Protocol commands, refer to the *Catalyst 5000 Series Command Reference* publication.

Enabling Spanning-Tree Protocol

To enable Spanning-Tree Protocol, enter this command in privileged mode:

Task	Command
Enable Spanning-Tree Protocol.	**set spantree enable** [*vlan*]

After enabling Spanning-Tree Protocol, you see this display:

```
Console> (enable) set spantree enable 1
VLAN1 bridge spanning tree enabled
Console< (enable)
```

Disabling Spanning-Tree Protocol

To disable Spanning-Tree Protocol, enter this command in privileged mode:

Task	Command
Disable Spanning-Tree Protocol.	**set spantree disable** [*vlan*]

> **Note** In a Token Ring environment, if you disable Spanning-Tree Protocol for a TrBRF, then all TrCRFs with this TrBRF as a parent are set to the forwarding state.

Setting the Bridge Forward Delay

To set the bridge forward delay for a VLAN, enter this command in privileged mode.

Task	Command
Set the bridge forward delay.	**set spantree fwddelay** *delay* [*vlan*]

Configuring Spanning-Tree Protocol for a TrBRF

You can configure the type of Spanning-Tree Protocol to be used by a TrBRF. Note that the following Spanning-Tree Protocol and bridge mode configurations are incompatible and can place logical ports in a blocked state:

- TrBRF is running the IBM Spanning-Tree Protocol and the TrCRF is in SRT mode.

- TrBRF is running the IEEE Spanning-Tree Protocol and the TrCRF is in SRB mode.

For more information, see the "Setting the Spanning-Tree Port State" section.

To specify a Spanning-Tree Protocol for a TrBRF, enter this command in privileged mode:

Task	Command
Specify a Spanning-Tree Protocol for a TrBRF.	**set vlan** vlan_num [**stp** {**ieee**	**ibm**}]

After entering the **set vlan** command and specifying a Spanning-Tree Protocol, you see this display:

```
Console> (enable) set vlan 950 stp ieee
Vlan 950 configuration successful
Console> (enable) show vlan 950
```

Specifying the Spanning-Tree Protocol Functional Address for a TrBRF

The **set spantree** command allows you to use the bridge functional address instead of the IEEE Spanning-Tree Protocol address when a TrBRF is configured to use the IEEE Spanning-Tree Protocol.

To specify that a TrBRF running the IEEE Spanning-Tree Protocol uses the bridge functional address instead of the IEEE Spanning-Tree Protocol address, enter this command in privileged mode:

Task	Command
Specify that a TrBRF running the IEEE Spanning-Tree Protocol use the bridge functional address instead of the IEEE Spanning-Tree Protocol address.	**set spantree multicast-address** *vlan_num* **ibm**

When you enable Spanning-Tree Protocol, every switch in the network goes through the blocking state and the transitory states at power up. If properly configured, the logical ports then stabilize to the forwarding or blocking state. However, with TrBRFs and TrCRFs, two exceptions require you to manually set the state of a logical port of a TrBRF:

- TrBRF is running the IBM Spanning-Tree Protocol and the TrCRF is in SRT mode.

- TrBRF is running the IEEE Spanning-Tree Protocol and the TrCRF is in SRB mode.

Note When the above configurations occur, the logical ports are put in a blocked state and no Spanning-Tree Protocol is run.

Setting the Spanning-Tree Port State

You can use the **set spantree portstate** command to manually set the state of a logical port to blocked or forwarding mode if either of the above configurations exists.

To manually set the state of a logical port, enter this command in privileged mode:

Task	Command
Manually set the state of a logical port.	**set spantree portstate** *trcrf* {**auto** | **block** | **forward**} [*trbrf*]

After entering the **set spantree portstate** command, you see this display:

```
Console> (enable) set spantree portstate 950 forward
Portstate successfully set for tokenring crf 950
Console> (enable)
```

Note If you disable the Spanning-Tree Protocol state for a TrBRF using the **set spantree** command, the logical ports of the TrBRF are put in forwarding state regardless of the state you configured using the **set spantree portstate** command.

Configuring the Port Priority

The port (physical or logical) with the lowest priority value has the highest priority and forwards the spanning-tree frames. The possible priority range is 0 through 255 (decimal). The default is 128. If all ports have the same priority value, the lowest port number forwards the spanning-tree frames.

To configure the priority associated with a logical port, enter this command in privileged mode:

Task	Command
Configure the priority associated with a logical port.	**set spantree portpri** *vlan_num priority*

After entering the **set spantree portpri** command, you see this display:

```
Console> (enable) set spantree portpri 900 50
Token Ring 900 port priority set to 50.
Console> (enable)
```

To configure the priority associated with a physical port, enter this command in privileged mode:

Task	Command
Configure the priority associated with a physical port.	**set spantree portpri** *mod_num/port_num priority*

After entering the **set spantree portpri** command, you see this display:

```
Console> (enable) set spantree portpri 3/4 10
Bridge port 3/4 port priority set to 10.
Console> (enable)
```

Configuring the Port Cost

The Spanning-Tree Protocol uses port path costs to determine which port (physical or logical) to select as a forwarding port. You should assign lower numbers to ports attached to faster media (such as full duplex), and higher numbers to ports attached to slower media. The possible range is 1 to 65535. The default is 62. Path cost is 1000 LAN speed in megabits per second.

To configure the cost associated with a physical port, enter this command in privileged mode:

Task	Command
Configure the cost associated with a physical port.	**set spantree portcost** *mod_num/port_num cost*

To configure the cost for a logical port, enter this command in privileged mode:

Task	Command
Configure the cost for a logical port.	**set spantree portcost** *trcrf cost*

After entering the **set spantree portcost** command and specifying a module number and port number, you see this display:

```
Console> (enable) set spantree portcost 3/4 100
Spantree port 3/4 path cost set to 100.
Console> (enable)
```

Configuring Spanning-Tree PortFast

You can configure a physical port that is connected to a single workstation or PC to start faster when it is connected, by entering this command in privileged mode:

Task	Command
Configure a physical port that is connected to a single workstation or PC to start faster when it is connected.	**set spantree portfast** *mod_num/port_num* {**enable**	**disable**}

Configuring Additional Spanning-Tree Protocol Parameters

You can configure additional Spanning-Tree Protocol parameters using the **set spantree** command.
Table 10-1 lists the command you use.

Table 10-1   Spanning Tree Configuration Commands

Task	Command
Set the bridge forward delay for a VLAN.	**set spantree fwddelay** *delay* [*vlan*]
Set the bridge hello time for a VLAN	**set spantree hello** *interval*
Set the bridge maximum aging time for a VLAN.	**set spantree maxage agingtime** [*vlan*]
Set the bridge priority for a VLAN.	**set spantree priority** *bridge_priority* [*vlan*]
Configure a physical port that is connected to a single workstation or PC to start faster when it is connected.	**set spantree portfast** *mod_num*/*port_num* {**enable** | **disable**}

Verifying Spanning-Tree Protocol Parameters

You can display the Spanning-Tree Protocol configuration for a specific VLAN, switching module,
or port. To verify the Spanning-Tree Protocol configuration information for a TrBRF or TrCRF, enter
this command in privileged mode:

Task	Command
Verify the Spanning-Tree Protocol configuration information.	**show spantree** *vlan_num*

After entering the **show spantree** command, you see this display:

```
Console> (enable) show spantree 1003
VLAN 1003
Spanning tree enabled

Designated Root            00-e0-1e-2f-6f-ea
Designated Root Priority   32768
Designated Root Cost       0
Designated Root Port       1/0
Root Max Age   20 sec    Hello Time 2  sec   Forward Delay 15 sec

Bridge ID MAC ADDR         00-e0-1e-2f-6f-ea
Bridge ID Priority         32768
Bridge Max Age 20 sec    Hello Time 2  sec   Forward Delay 15 sec

 Port      Vlan  Port-State    Cost   Priority  Fast-Start  Group-method
 --------- ----  ------------- -----  --------  ----------  ------------
  3/1      1003  not-connected  250        32   disabled
  3/2      1003  not-connected  250        32   disabled
  3/3      1003  not-connected  250        32   disabled
  3/4      1003  not-connected  250        32   disabled
  .
  .
  .
  3/12     1003  not-connected  250        32   disabled
  3/13     1003  not-connected  250        32   disabled
  3/14     1003  not-connected  250        32   disabled
  3/15     1003  not-connected  250        32   disabled
  3/16     1003  not-connected  250        32   disabled
Console> (enable)
```
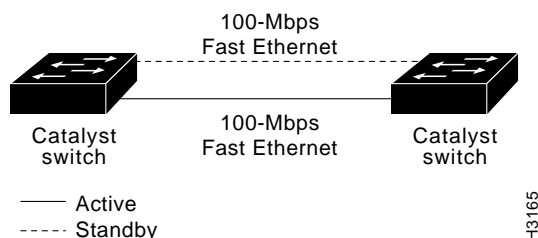
Spanning-Tree Examples

You can design fault-tolerant connections using Ethernet only or Ethernet combined with other topologies. Figure 10-1 and Figure 10-2 show fault-tolerant Fast Ethernet topologies using the Spanning-Tree Protocol.

Figure 10-1   Fault-Tolerant Fast Ethernet Topology Example

Figure 10-2   Fault-Tolerant Fast Ethernet Topology with Increased Capacity Example
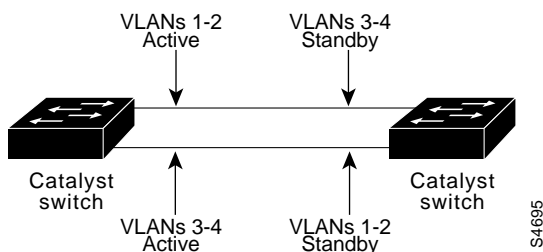
Understanding Spanning-Tree Protocol

Spanning-Tree Protocol is a link management protocol that provides path redundancy while preventing undesirable loops in the network. For an Ethernet network to function properly, only one active path must exist between two stations.

Multiple active paths between stations cause loops in the network. If a loop exists in the network, you might receive duplicate messages. When loops occur, some switches see stations on both sides of the switch. This condition confuses the forwarding algorithm and allows duplicate frames to be forwarded.

To provide path redundancy, Spanning-Tree Protocol defines a tree that spans all switches in an extended network. Spanning-Tree Protocol forces certain redundant data paths into a standby (blocked) state. If one network segment in the Spanning-Tree Protocol becomes unreachable, or if Spanning-Tree Protocol costs change, the spanning-tree algorithm reconfigures the spanning-tree topology and reestablishes the link by activating the standby path.

Spanning-Tree Protocol operation is transparent to end stations, which are unaware whether they are connected to a single LAN segment or a switched LAN of multiple segments.

Election of the Root Switch

All switches in an extended LAN participating in Spanning-Tree Protocol gather information on other switches in the network through an exchange of data messages called Bridge Protocol Data Units (BPDUs). This exchange of messages results in the following actions:

- The election of a unique root switch for the stable spanning-tree network topology.
- The election of a designated switch for every switched LAN segment.
- The removal of loops in the switched network by placing redundant switch ports in a backup state.

The Spanning-Tree Protocol root switch is the logical center of the spanning-tree topology in a switched network. All paths that are not needed to reach the root switch from anywhere in the switched network are placed in Spanning-Tree Protocol backup mode. Table 10-2 describes the root switch variables that affect the entire spanning-tree performance.

Table 10-2   Spanning-Tree Protocol Root Switch Variables

Variable	Description
Hello Time	Determines how often the switch broadcasts its hello message to other switches.
Maximum Age Timer	Measures the age of the received protocol information recorded for a port and ensures that this information is discarded when its age limit exceeds the value of the maximum age parameter recorded by the switch. The timeout value for this timer is the maximum age parameter of the switches.
Forward Delay Timer	Monitors the time spent by a port in the learning and listening states. The timeout value is the forward delay parameter of the switches.

BPDUs contain information about the transmitting switch and its ports, including switch and port Media Access Control (MAC) addresses, switch priority, port priority, and port cost. The Spanning-Tree Protocol uses this information to elect the root switch and root port for the switched network, as well as the root port and designated port for each switched segment.

Figure 10-3 shows how BPDUs enable a Spanning-Tree Protocol topology.

Figure 10-3   BPDUs Enabling a Stable Spanning-Tree Protocol Topology
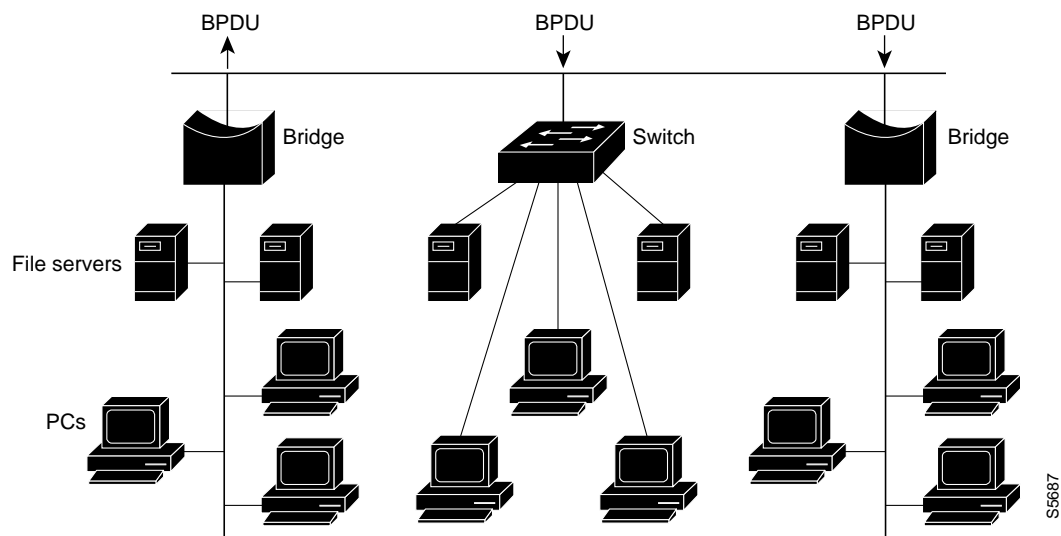
Bridge Protocol Data Units

The stable active topology of a switched network is determined by the following:

- The unique switch identifier (MAC address) associated with each switch.
- The path cost to the root associated with each switch port.
- The port identifier (MAC address) associated with each switch port.

Each configuration BPDU contains the following minimal information:

- The unique identifier of the switch that the transmitting switch believes to be the root switch.
- The cost of the path to the root from the transmitting port.
- The identifier of the transmitting port.

The switch sends configuration BPDUs to communicate and compute the spanning-tree topology. A MAC frame conveying a BPDU sends the switch group address to the destination address field. All switches connected to the LAN on which the frame is transmitted receive the BPDU. BPDUs are not directly forwarded by the switch, but the receiving switch uses the information in the frame to calculate a BPDU, and, if topology changes, instigate a BPDU transmission.

A BPDU exchange results in the following:

- One switch is elected as the root switch.
- The shortest distance to the root switch is calculated for each switch.
- A designated switch is selected. This is the switch closest to the root switch through which frames will be forwarded to the root.
- A port for each switch is selected. This is the port providing the best path from the switch to the root switch.
- Ports included in the Spanning-Tree Protocol are selected.

Spanning-Tree Protocol Configuration

If all switches are enabled with default settings, the switch with the lowest MAC address in the network becomes the root switch. The network in Figure 10-4 assumes that Switch A has the lowest MAC address and is therefore the root switch. However, due to traffic patterns, number of forwarding ports, or line types, Switch A might not be the ideal root switch. By increasing the priority (lowering the numerical priority number) of the ideal switch so that it becomes the root switch, you force a Spanning-Tree Protocol recalculation to form a new, stable topology.

Figure 10-4   Configuring a Stable Topology

RP = Root Port
DP = Designated Port
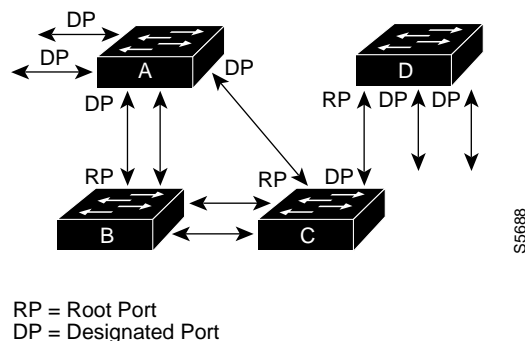
When the stable Spanning-Tree Protocol topology is based on default parameters, the path between source and destination stations in a switched network might not be the most ideal. For instance, connecting higher-speed links to a port that has a higher number than the current root port can cause a root-port change. The point is to make the fastest link the root port.

For example, assume that port 2 on Switch B in Figure 10-4 is a fiber-optic link, and that port 1 on Switch B (a UTP link) is the root port. Network traffic might be more efficient over the high-speed fiber-optic link. By changing the Port Priority parameter for port 2 to a higher priority (lower numerical value) than port 1, port 2 becomes the root port. The same change can occur by changing the Port Cost parameter for port 2 to a lower value than that of port 1.

Spanning-Tree Protocol Port States

Propagation delays can occur when protocol information passes through a switched LAN. As a result, topology changes can take place at different times and at different places in a switched network. When a switch port transitions directly from nonparticipation in the stable topology to the forwarding state, it can create temporary data loops. Ports must wait for new topology information to propagate through the switched LAN before starting to forward frames. They must allow the frame lifetime to expire for frames that have been forwarded using the old topology.

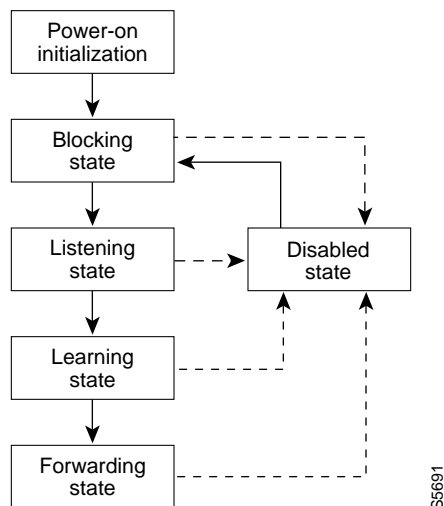Each port on a switch using Spanning-Tree Protocol exists in one of the following five states:

- Blocking

- Listening

- Learning

- Forwarding

- Disabled

A port moves through these five states as follows:

- From initialization to blocking

- From blocking to listening or to disabled

- From listening to learning or to disabled

- From learning to forwarding or to disabled

- From forwarding to disabled

Figure 10-5 illustrates how a port moves through the five states.

Figure 10-5   **Spanning-Tree Protocol Port States**

You can modify each port state by using management software. When you enable Spanning-Tree Protocol, every switch in the network goes through the blocking state and the transitory states of listening and learning at power up. If properly configured, the ports then stabilize to the forwarding or blocking state.

When the spanning-tree algorithm places a port in the forwarding state, the following occurs:

- The port is put into the listening state while it waits for protocol information that suggests it should go to the blocking state.

- The port waits for the expiration of a protocol timer that moves the port to the learning state.

- In the learning state, the port continues to block frame forwarding as it learns station location information for the forwarding database.

- The expiration of a protocol timer moves the port to the forwarding state, where both learning and forwarding are enabled.

Blocking State

A port in the blocking state does not participate in frame forwarding, as shown in Figure 10-6. After initialization, a BPDU is sent to each port in the switch. A switch initially assumes it is the root until it exchanges BPDUs with other switches. This exchange establishes which switch in the network is really the root. If only one switch resides in the network, no exchange occurs, the forward delay timer expires, and the ports move to the listening state. A switch always enters the blocking state following switch initialization.

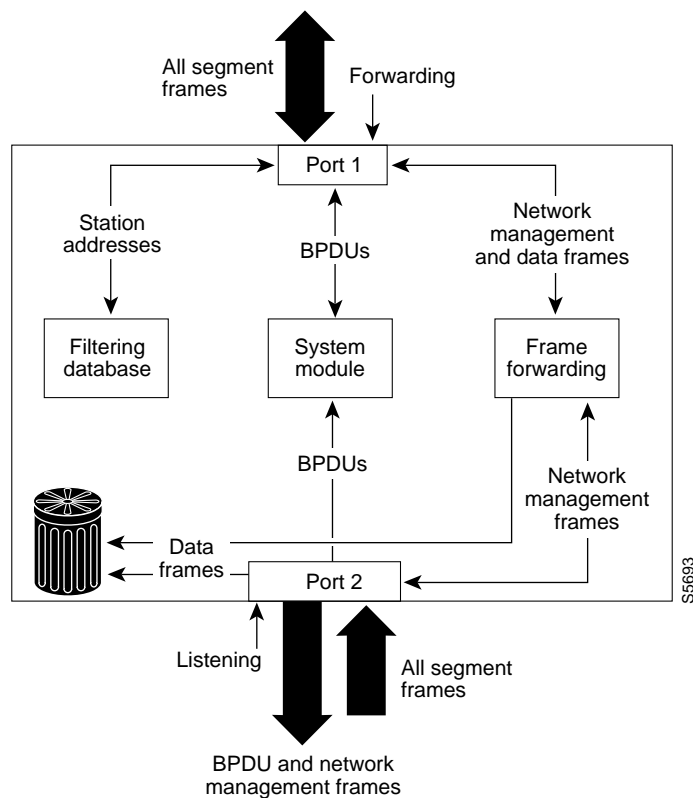**Figure 10-6   Port 2 in Blocking State**
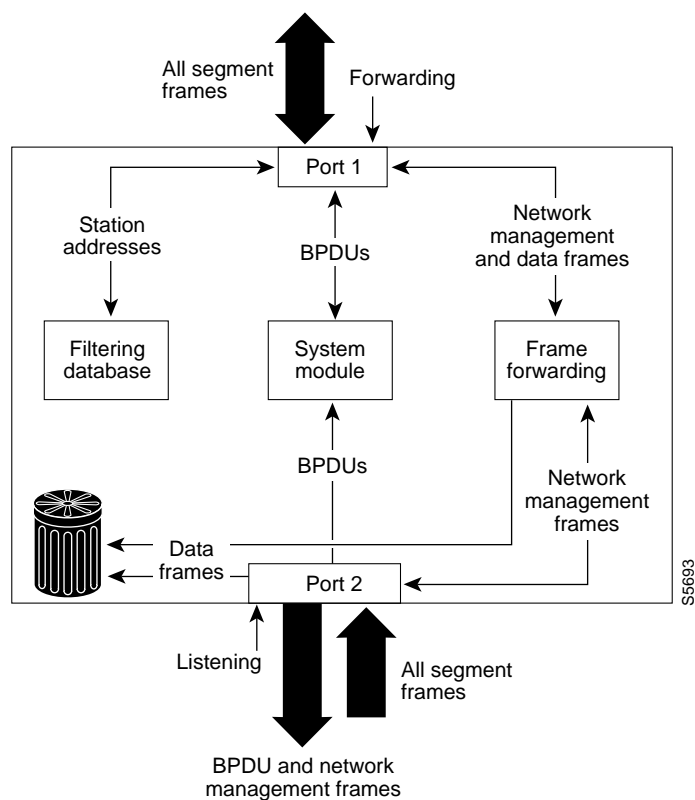
A port in the blocking state performs as follows:

- Discards frames received from the attached segment.

- Discards frames switched from another port for forwarding.

- Does not incorporate station location into its address database. (There is no learning at this point, so there is no address database update.)

- Receives BPDUs and directs them to the system module.

- Does not transmit BPDUs received from the system module.

- Receives and responds to network management messages.

Listening State

The listening state is the first transitional state a port enters after the blocking state, when Spanning-Tree Protocol determines that the port should participate in frame forwarding. Learning is disabled in the listening state. Figure 10-7 shows a port in the listening state.

Figure 10-7   Port 2 in Listening State

A port in the listening state performs as follows:

- Discards frames received from the attached segment.

- Discards frames switched from another port for forwarding.

- Does not incorporate station location into its address database. (There is no learning at this point, so there is no address database update.)

- Receives BPDUs and directs them to the system module.

- Processes BPDUs received from the system module.

- Receives and responds to network management messages.

Learning State

A port in the learning state prepares to participate in frame forwarding. The port enters the learning state from the listening state.
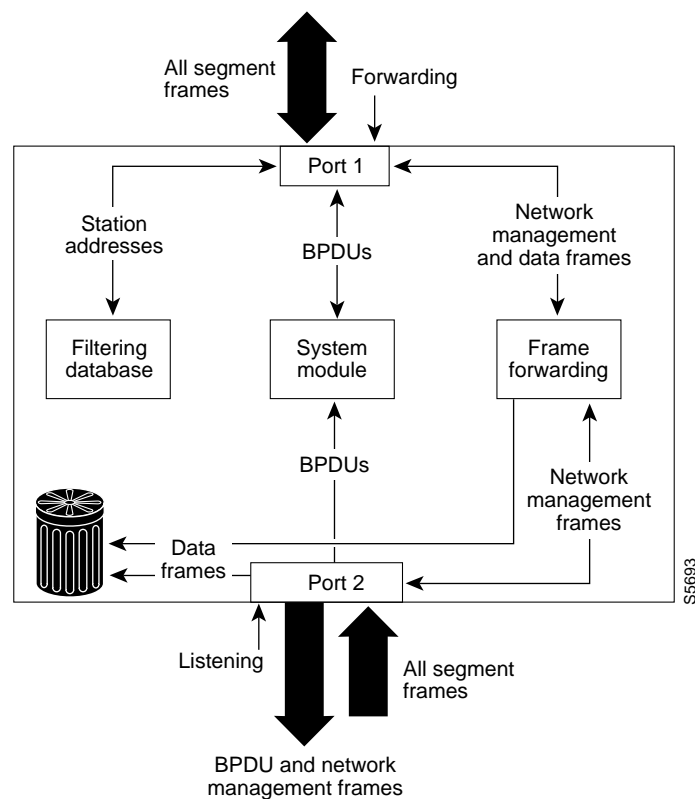
A port in the learning state performs as follows:

- Discards frames received from the attached segment.

- Discards frames switched from another port for forwarding.

- Incorporates station location into its address database.

- Receives BPDUs and directs them to the system module.

- Receives, processes, and transmits BPDUs received from the system module.

- Receives and responds to network management messages.

Forwarding State

A port in the forwarding state forwards frames, as shown in Figure 10-8. The port enters the forwarding state from the learning state.

Figure 10-8   Port 2 in Forwarding State

A port in the forwarding state performs as follows:

- Forwards frames received from the attached segment.

- Forwards frames switched from another port for forwarding.

- Incorporates station location information into its address database.
- Receives BPDUs and directs them to the system module.
- Processes BPDUs received from the system module.
- Receives and responds to network management messages.

⚠ **Caution** Use the immediate-forwarding (**portfast**) mode only on ports connected to individual workstations to allow these ports to come up and go directly to the forwarding state, rather than having to go through the entire spanning-tree initialization process. To prevent illegal topologies, enable Spanning-Tree Protocol on ports connected to switches or other devices that forward messages.

Disabled State

A port in the disabled state does not participate in frame forwarding or Spanning-Tree Protocol, as shown in Figure 10-9. A port in the disabled state is virtually nonoperational.

Figure 10-9   Port 2 in Disabled State
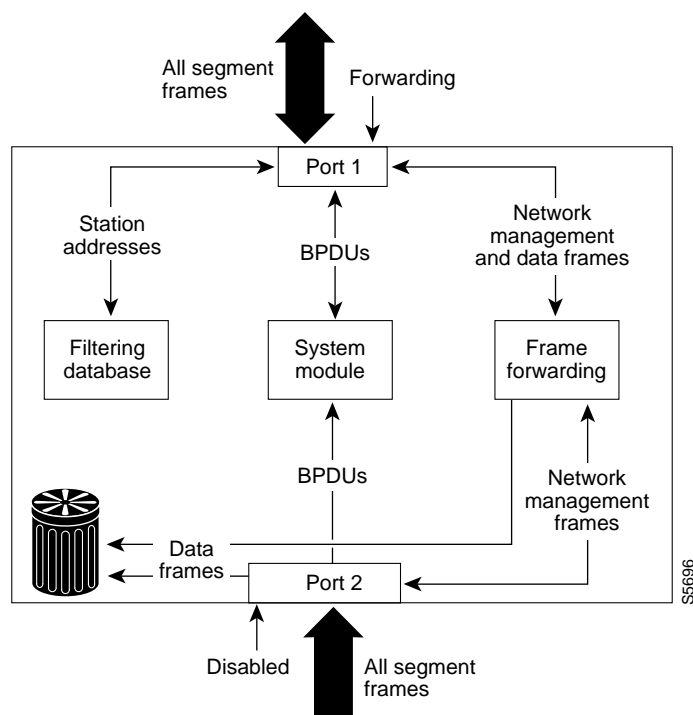
A disabled port performs as follows:

- Discards frames received from the attached segment.
- Discards frames switched from another port for forwarding.
- Does not incorporate station location into its address database. (There is no learning, so there is no address database update.)
- Receives BPDUs, but does not direct them to the system module.

- Does not receive BPDUs for transmission from the system module.

- Receives and responds to network management messages.

Understanding Spanning-Tree for Token Ring

Typically, each VLAN runs one instance of Spanning-Tree Protocol to prevent loops in the bridge topology. However, Token Ring runs Spanning-Tree Protocol both at the TrCRF level and the TrBRF level. The Spanning-Tree Protocol that runs at the TrCRF level removes loops in the logical ring. The TrBRF Spanning-Tree Protocol is similar to the Ethernet Spanning-Tree Protocol, interacting with external bridges to remove loops from the bridge topology.

The Catalyst 5000 series Token Ring module supports these Spanning-Tree Protocols:

- IEEE 802.1D Spanning-Tree Protocol

- IBM Spanning-Tree Protocol

- Cisco Spanning-Tree Protocol

The Catalyst 5000 series switch uses the IEEE 802.1D and IBM Spanning-Tree Protocols on TrBRFs. The Spanning-Tree Protocol that runs on the TrCRF is either the Cisco or IEEE Spanning-Tree Protocol, depending on the bridging mode you configured for the TrCRF with the **set vlan** command.

Caution   Certain TrBRF Spanning-Tree Protocol and TrCRF bridge mode configurations are incompatible and can place the TrCRFs in a blocked state. For more about these configurations, see the "Setting the Spanning-Tree Port State" section.

Configuring Spanning-Tree Uplink Fast Switchover

This section describes how to configure the Spanning-Tree Protocol Uplink Fast Switchover feature, which is also called *UplinkFast*.

Enabling UplinkFast

UplinkFast provides fast convergence after a spanning-tree topology change and achieves load balancing between redundant links using uplink groups. An uplink group is a set of ports (per VLAN), only one of which is forwarding at any given time. Specifically, an uplink group consists of the root port (which is forwarding) and a set of blocked ports, except for self-looping ports. The uplink group provides an alternate path in case the currently forwarding link fails.

Note   The UplinkFast feature is most useful in wiring-closet switches. This feature may not be useful for other types of applications.

Procedures for configuring UplinkFast appear in this section.

Configuring a Primary Root Switch

To configure a switch as the primary root switch, enter this command:

Task	Command
Configure a switch as the primary root switch.	**set spantree root** *vlans* [**dia** *network_diameter*] [**hello** *hello_time*]

After entering the command you see this display. This example shows how to specify a Catalyst 5000 series switch as the primary root switch for VLANs 1–10:

```
Console> (enable) set spantree root 1-10 dia 4
VLANs 1-10 bridge priority set to 8192
VLANs 1-10 bridge max aging time set to 14 seconds.
VLANs 1-10 bridge hello time set to 2 seconds.
VLANs 1-10 bridge forward delay set to 9 seconds.
Switch is now the root switch for active VLANs 1-6.
Console> (enable)
```

Note   Run the **set spantree root** command on backbone switches or distribution switches only, not on access switches.

The **set spantree root** command reduces the bridge priority (the value associated with the switch) from the default (32,768) to a significantly lower value, which allows the switch to become the root switch.

When you specify a switch to become the primary root, the default bridge priority is modified so that it becomes the root for the specified VLANs. Set the bridge priority to 8192. If this setting does not result in the switch becoming a root, modify the bridge priority to be 100 less than the bridge priority of the current root switch. Since different VLANs could potentially have different root switches, the bridge VLAN-priority chosen makes this switch the root for all the VLANs specified. If reducing the bridge priority as low as 1 still does not make the switch the root switch, the system displays a message.

Configuring a Secondary Root Switch

To configure a switch as the secondary root switch, enter this command:

Task	Command
Configure a switch as the secondary root switch.	**set spantree root** [**secondary**] *vlans* [**dia** *network_diameter*] [**hello** *hello_time*]

After you enter this command, you see this display. The following example shows how to specify a Catalyst 5000 series switch as the secondary root switch for VLANs 22 and 24:

```
Console> (enable) set spantree root secondary 22,24 dia 5 hello 1
VLANs 22,24 bridge priority set to 16384.
VLANs 22,24 bridge max aging time set to 10 seconds.
VLANs 22,24 bridge hello time set to 1 second.
VLANs 22,24 bridge forward delay set to 7 seconds.
Console> (enable)
```

The **set spantree root secondary** command reduces the bridge priority to 16,384, making it the probable candidate to become the root switch if the primary root switch fails. You can run this command on more than one switch to create multiple backup switches in case the primary root switch fails.

Enabling and Disabling UplinkFast

To enable and disable UplinkFast, perform these tasks:

Task	Command
Enable the UplinkFast feature.	**set spantree uplinkfast enable** [**rate** *station_update_rate*] [**all-protocols off**	**on**]
Disable the UplinkFast feature.	**set spantree uplinkfast disable**

This command increases the path cost of all ports on the switch, making it unlikely that the switch becomes the root switch. The *station_update_rate* value represents the number of multicast packets transmitted per 100 milliseconds (the default is 15 packets per millisecond).

After entering the **set spantree uplinkfast enable** command, you see this display. This example shows how to enable and verify the UplinkFast feature with a station-update rate of 40 packets per 100 milliseconds:

Note When you enable the **set spantree uplinkfast** command, it affects all VLANs on a Catalyst 5000 series switch. You cannot configure the UplinkFast feature on an individual VLAN.

```
Console> (enable) set spantree uplinkfast enable rate 40
VLANs 1-1000 bridge priority set to 49152.
The port cost and portvlancost of all ports increased by 3000.
Station update rate set to 40 packets/100ms.
uplinkfast turned on for bridge.
Console> (enable)

Console> (enable) show spantree uplinkfast
VLAN          port list
-----------------------------------------------
1-20          1/1(fwd),1/2-1/5
21-50         1/9(fwd), 1/6-1/8, 1/10-1/12
```

Assigning a Lower Cost to VLANs

To assign a lower cost to a set of VLANs on a port, perform this task:

Task	Command
Lower the cost of a set of VLANs on a port.	**set spantree portvlancost** *mod_num/port_num* [**cost** *cost_value*] [**preferred_vlans**]

If you do not specify the VLANs, the command acts on the VLANs specified in prior instances of this command. If you do not specify a cost, the **portvlancost** value is set to one less than the current **portcost** value for the port. This example shows how to assign a lower cost to a set of VLANs on a port:

```
Console> (enable) set spantree portvlancost 2/10 1-20
Port 2/10 VLANs 1-20 have a path cost of 9.
```

To achieve load balancing between redundant links, you can modify the path cost of the uplink ports on a per-VLAN basis using the **set spantree portvlancost** command. To determine the root path cost for a switch through a particular port, the path cost of the port is added to the root path cost defined in the configuration BPDU. You can also achieve load balancing by designating different backbone switches as the root switch for different VLANs.

For descriptions of the commands used to configure the Spanning-Tree Uplink Fast Switchover feature, refer to the *Catalyst 5000 Series Command Reference* publication.

Understanding Uplink Groups

Switches are normally connected hierarchically, as shown in Figure 10-10. In simpler networks, the upper two levels of the hierarchy may be collapsed into a single backbone layer. Figure 10-10 shows the network topology after spanning tree converges into a loop-free topology. Spanning tree has blocked the redundant links to avoid loops. Every access switch and distribution switch in the figure has a redundant uplink.
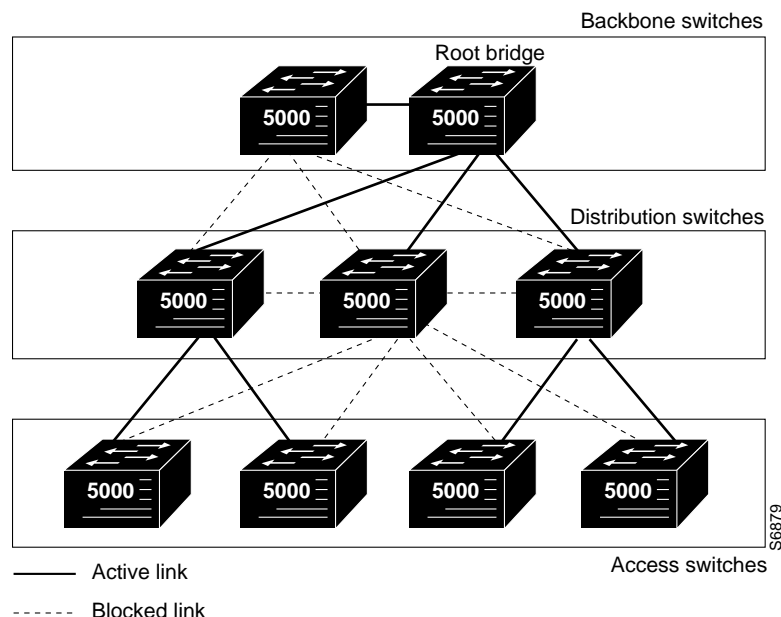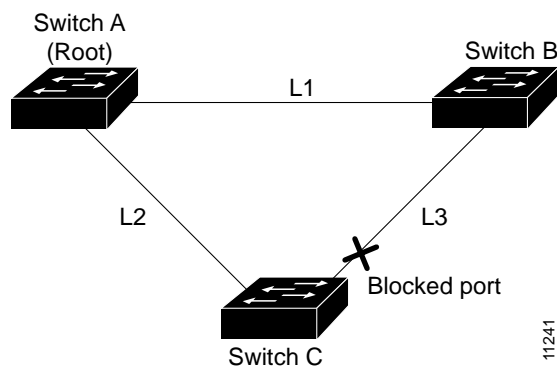
Figure 10-10   Hierarchical Switch Topology

Figure 10-11 shows an example topology with no link failures. Switch A, the root switch, is connected directly to Switch B over link L1 and to Switch C over link L2. The port on Switch C that is connected directly to Switch B is in blocking state.

Figure 10-11   UplinkFast Example Before Direct Link Failure

If Switch C detects a link failure on the currently active link L2 (a *direct* link failure), UplinkFast unblocks the blocked port on Switch C and transitions it to the forwarding state without going through the listening and learning states, as shown in Figure 10-12. This switchover takes approximately one to five seconds.

Figure 10-12   UplinkFast Example After Direct Link Failure
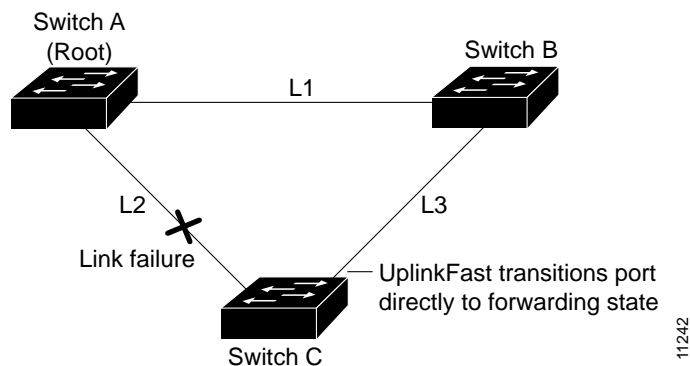
Configuring Spanning-Tree Protocol Backbone Fast Convergence

This section describes how to configure the Spanning-Tree Backbone Fast Convergence feature, also known as the *BackboneFast* feature.

Enabling BackboneFast Convergence

The BackboneFast Convergence feature reduces the time needed for the spanning tree to converge after experiencing a topology change caused by indirect link failures. This feature complements the UplinkFast feature described in the previous section. However, the Backbone Fast Convergence feature is designed for all switches that experience indirect link failures.

Note   For the Backbone Fast feature to work, you must enable it on all switches in the network. The Backbone Fast feature is not supported on Token Ring VLANs. This feature is supported for use with third-party switches.

Procedures

This section contains procedures for configuring BackboneFast Convergence.

Enabling BackboneFast

To configure the Backbone Fast Convergence feature, enter this command:

Task	Command
Enable the Backbone Fast Convergence feature.	**set spantree backbonefast enable**

After enabling Backbone Fast Convergence, you see this display:

```
Console> (enable) set spantree backbonefast enable
Backbonefast enabled for all VLANs.
```

Displaying a Summary of Connected Spanning-Tree Ports

To display a summary of connected spanning-tree ports by VLAN, enter this command:

Task	Command
Display a summary of connected spanning-tree ports.	**show spantree summary**

After entering the show spantree summary command, you see this display:

```
Console> (enable) show spantree summary
Summary of connected spanning tree ports by vlan

Uplinkfast disabled for bridge.
Backbonefast enabled for bridge.

Vlan  Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
    1        0         0        0          1          1

      Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
Total        0         0        0          1          1

BackboneFast statistics
-----------------------
Number of inferior BPDUs received (all VLANs)  : 0
Number of RLQ req PDUs received (all VLANs)    : 0
Number of RLQ res PDUs received (all VLANs)    : 0
Number of RLQ req PDUs transmitted (all VLANs) : 0
Number of RLQ res PDUs transmitted (all VLANs) : 0
Console> (enable)
```

Disabling BackboneFast

To disable the BackboneFast feature, enter this command:

Task	Command
Disable the BackboneFast feature.	**set spantree backbonefast disable**

Verification

To verify that the BackboneFast feature is enabled, enter this command:

```
Console> (enable) show spantree backbonefast
Backbonefast is enabled.
```

To verify that the BackboneFast feature is disabled, enter this command:

```
Console> (enable) show spantree backbonefast
Backbonefast is disabled.
```

Understanding Backbone Fast Convergence

The BackboneFast feature is initiated when a root port or blocked port on a switch receives inferior BPDUs from its designated bridge. An inferior BPDU is a BPDU that identifies one switch as both the root bridge and the designated bridge. When a switch receives an inferior BPDU, it indicates that a link to which the switch is not directly connected (an *indirect* link) has failed (that is, the designated bridge has lost its connection to the root bridge). Under normal spanning-tree rules, the switch ignores inferior BPDUs for the configured maximum aging time, as specified by the *agingtime* variable of the **set spantree maxage** command.

The switch tries to determine if it has an alternate path to the root bridge. If the inferior BPDU arrives on a blocked port, the root port and other blocked ports on the switch become alternate paths to the root bridge. (Self-looped ports are not considered alternate paths to the root bridge.) If the inferior BPDU arrives on the root port, all blocked ports become alternate paths to the root bridge. If the inferior BPDU arrives on the root port and there are no blocked ports, the switch assumes that it has lost connectivity to the root bridge, causes the maximum aging time on the root to expire, and becomes the root switch according to normal spanning-tree rules.

If the switch has alternate paths to the root bridge, it uses these alternate paths to transmit a new kind of PDU called the "Root Link Query" PDU. The switch sends the Root Link Query PDU out all alternate paths to the root bridge. If the switch determines that it still has an alternate path to the root, it causes the maximum aging time on the ports on which it received the inferior BPDU to expire. If all the alternate paths to the root bridge indicate that the switch has lost connectivity to the root bridge, the switch causes the maximum aging times on the ports on which it received an inferior BPDU to expire. If one or more alternate paths can still connect to the root bridge, the switch makes all ports on which it received an inferior BPDU its designated ports and moves them out of the blocking state (if they were in blocking state), through the listening and learning states, and into the forwarding state.

Figure 10-13 shows an example topology with no link failures. Switch A, the root switch, connects directly to Switch B over link L1 and to Switch C over link L2. The port on Switch C that connects directly to Switch B is in the blocking state.

Figure 10-13   BackboneFast Example Before Indirect Link Failure
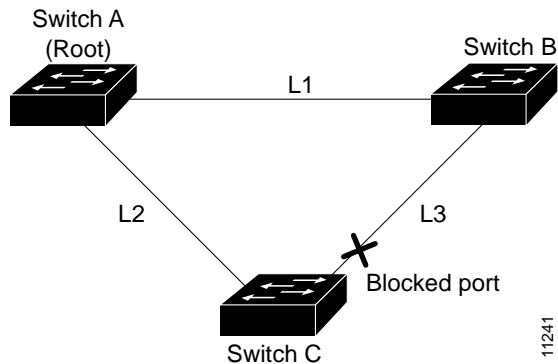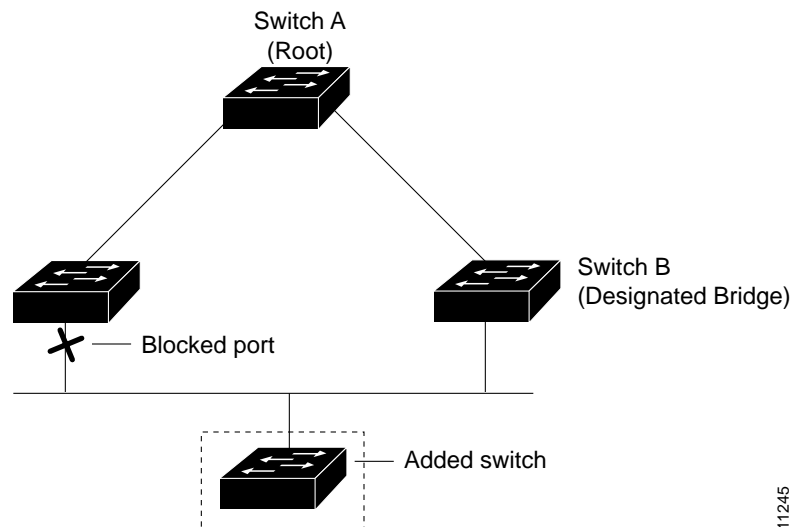

If link L1 fails, Switch C detects this failure as an indirect failure, since it is not connected directly to link L1. Switch B no longer has a path to the root switch. The BackboneFast feature allows the blocked port on Switch C to move immediately to the listening state without waiting for the maximum aging time for the port to expire. The BackboneFast feature then transitions the port on Switch C to the forwarding state, providing a path from Switch B to Switch A. This switchover takes approximately 30 seconds. Figure 10-14 shows how the BackboneFast feature reconfigures the topology to account for the failure of link L1.

Figure 10-14   BackboneFast Example After Indirect Link Failure
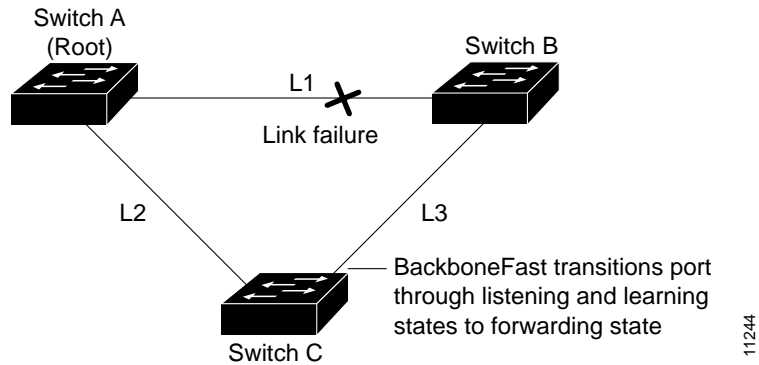
If a new switch is introduced into a shared-medium topology, the BackboneFast feature is not activated. Figure 10-15 shows a shared-medium topology in which a new switch is added. The new switch begins sending inferior BPDUs that say it is the root switch. However, the other switches ignore these inferior BPDUs and the new switch learns that Switch B is the designated bridge to Switch A, the root switch.

Figure 10-15   Adding a Switch in a Shared-Medium Topology